INFORMATION AND COMPUTATION **132**, 85–108 (1997)
ARTICLE NO. IC962616

Approximation Algorithms for

Klaus Jansen

Fachbereich IV-Mathematik, Universität Trier, Universitätsring 15, 54286 Trier, Germany
E-mail: jansen@dm3.uni-trier.de

and

Sabine Öhring

Department of Computer Science, University of North Texas, Denton, Texas, 76203-3886
E-mail: oehring@cs.unt.edu

In this paper we consider the following time constrained scheduling problem. Given a set of jobs J with execution times $e(j) \in (0, 1]$ and an undirected graph $G=(J, E)$, we consider the problem to find a schedule for the jobs such that adjacent jobs $(j, j') \in E$ are assigned to different machines and that the total execution time for each machine is at most 1. The goal is to find a minimum number of machines to execute all jobs under this time constraint. This scheduling problem is a natural generalization of the classical bin-packing problem. We propose and analyse several approximation algorithms with constant absolute worst case ratio for graphs that can be colored in polynomial time.

1. INTRODUCTION

Let J be a set of jobs with execution times $e(j) \in \mathbb{R}^+$ $(0 < e(j) \leqslant 1)$ and let $G=(J, E)$ be a conflict graph. We look for a partition of the job set J into independent sets $U_1, ..., U_m$ such that $\sum_{j \in U_i} e(j) \leqslant 1$ for each $1 \leqslant i \leqslant m$. For each edge $e=\{j, j'\} \in E$ the corresponding jobs j and j' must be processed on different machines. Each independent set U_i has to be executed by one machine with total execution time at most 1. The goal of the studied problem is to find such a partition with a minimum number m of machines.

One application of the problem is the assignment of processes to processors. In this case, we have a set of processes (e.g., multimedia streams) where some of the processes are not allowed to execute on the same processor. This can be for reason of fault tolerance (not to schedule two replicas of the same process on the same cabinet) or for efficiency purposes (better put two cpu intensive processes on different processors). The problem is how to assign a minimum number of processors

0890-5401/97 $25.00

for this set of processes. A second application is given by storing versions of the same file or a database. Again, for reason of fault tolerance we would like to keep two replicas/versions of the same file on different file server.

A complementary problem has been investigated by Baker and Coffman [1] and by Irani and Leung [12]. They study the problem of scheduling jobs constrained by a mutual exclusion graph G where jobs that are in conflict must be executed one after another. Each job requires one unit of running time and is executed on one processor. The goal of the complementary problem is to find a schedule with minimum total completion time. If $e(j)$ gives the amount of a given resource (e.g., storage, input–output device), then we obtain our orginal problem with another interpretation.

If E is an empty set, this scheduling problem is equal to the classical bin-packing problem. Furthermore, if $\sum_{j \in J} e(j) \leqslant 1$ then we obtain the problem to compute the chromatic number $\chi(G)$ of the conflict graph G. This means that the time constrained scheduling problem is NP-complete even if $E = \emptyset$ or if $\sum_{j \in J} e(j) \leqslant 1$.

We propose methods which generate approximate solutions for the problem in polynomial time. We define ω^H to be the number of machines when the jobs are assigned by a heuristic H, and ω^* to be the corresponding optimum number. If independently of the problem instances, $\omega^H \leqslant \rho \omega^*$ holds for a specified constant ρ, with ρ as small as possible, then ρ is called the absolute worst case performance ratio of heuristic H. The approximation algorithms we propose in this paper work for conflict graphs belonging to graph classes that are easy (i.e., in polynomial time) to color.

We notice that no polynomial time algorithm has an absolute worst case ratio smaller than 1.5 for the bin-packing or the considered scheduling problem, unless $P = NP$. This is obvious since such an algorithm could be used to solve the partition problem [8] in polynomial time. The scheduling problem for an arbitrary undirected graph is harder to approximate, because Lund and Yannakakis [16] recently proved that unless $P = NP$ there is an $\varepsilon > 0$ such that no polynomial time approximation algorithm for the coloring problem can guarantee a worst case ratio better than $|J|^\varepsilon$.

The orthogonal problem, called resource constrained scheduling, where the number of machines m is given and where the maximum completion time (called makespan) is minimized was studied before in [3]. In [3] an approximation algorithm is given for the case that we have a priori a k-coloring of the conflict graph. The algorithm has the worst case ratio $(k+2)/2$ for $m \geqslant k+1$, and when m/k tends to infinity the worst case ratio tends to 2. Moreover, Bodlaender *et al.* [3] proved that unless $P = NP$ no approximation algorithm can beat the worst case ratio 2.

Furthermore, in [4] the resource constrained scheduling problem with unit-values ($e(j) = 1/\ell$ for each job $j \in J$) was studied. In [4], the computational complexity of this problem is studied for different graph classes like bipartite graphs, interval graphs and cographs, arbitrary and constant numbers m and ℓ. Baker and Coffman [1] have proved, e.g., that forests can be scheduled optimally in polynomial time and have investigated scheduling of graphs resulting from a two-dimensional domain decomposition problem. Furthermore, Irani and Leung [12] have studied on-line algorithms for interval and bipartite graphs. These graphs are motivated by traffic intersection control and scheduling in high speed local area networks.

In this paper, we study seven approximation algorithms for the time constrained problem. The first algorithm is a direct modification of bin-packing heuristics. In Section 2, it is shown that the first algorithm does not have a constant worst case ratio. The other algorithms are based on the composition of two algorithms—a coloring algorithm and bin-packing heuristics applied on a part of the conflict graph. The simplest algorithm given in Section 3 uses an optimum coloring and the next-fit heuristic and has the absolute worst case ratio 3. For first-fit the bound becomes 2.7 and for first-fit decreasing the bound lies between 2.691 and 2.7. Applied to bipartite graphs the approximation algorithm generates the ratio 2.

The third algorithm is based on a precoloring method. The main step is to compute a minimum coloring of the conflict graph where the long jobs are separated or colored differently. Based on this method, we obtain an approximation algorithm with worst case ratio $\frac{5}{2}$ for graphs like interval graphs, split graphs, cographs and further graphs; see Section 4. More involved coloring methods are studied in Section 5. We obtain approximation algorithms with worst case bounds $\frac{7}{3}$, $\frac{11}{5}$ and $\frac{15}{7}$. The last algorithm is a general separation method that works for cographs and partial K-trees. Based on this separation method we obtain an approximation algorithm with worst case ratio $2+\varepsilon$ for these two graph classes. This result implies approximation with factor $2+\varepsilon$ for any class of graphs with a constant upper bound on the treewidth (e.g., outerplanar graphs, series parallel graphs and Halin graphs).

2. DIRECT BINPACKING

In this section, we analyse an algorithm that is a direct modification of the bin-packing heuristics. For a survey on the bin-packing problem we refer to [6]. First, we describe the bin-packing problem and three important approximation algorithms.

The bin-packing problem is stated as follows [14]. Let $L=(a_1, ..., a_n)$ be a list of real positive numbers a_i $(0<a_i\leqslant 1)$. The problem is to place the elements of L into a minimum number $\omega^*(L)$ of bins such that no bin contains numbers whose sum exceeds the capacity 1. We index the bins as $B_1, B_2, ...,$ initially filled to level zero, and we place the numbers $a_1, ..., a_n$ in the given order of the list L. The well-known bin-packing heuristics *NF* (Next Fit), *FF* (First Fit), and *FFD* (First Fit Decreasing) can be described as follows:

Algorithm (*NF*). To place a_i, take the highest index j such that B_j is filled to level $\beta>0$. If $\beta+a_i\leqslant 1$ then place a_i into B_j; otherwise place a_i into the next empty bin B_{j+1}.

Algorithm (*FF*). To place a_i, find the smallest j such that B_j is filled to level $\beta\leqslant 1-a_i$ and place a_i into B_j.

Algorithm (*FFD*). Arrange the list $L=(a_1, ..., a_n)$ into non-increasing order and apply the algorithm (*FF*) on the derived list.

It is not difficult to show that the number of bins used in the Next Fit packing $\omega^{NF}(L)$ is at most $2\omega^*(L)$. The best bound on the performance of *FF* is given in [7] where they show that

$$\omega^{FF}(L) \leqslant \lceil \tfrac{17}{10}\omega^*(L) \rceil.$$

Currently, the best bound on the performance of *FFD* has been obtained by Minyi [17] who shows that

$$\omega^{FFD}(L) \leqslant \tfrac{11}{9}\omega^*(L) + 1.$$

Therefore, the asymptotic worst case ratio of the *FF* heuristic is $\frac{17}{10}$ while *FFD* has an asymptotic worst case ratio of $\frac{11}{9}$. Simchi–Levi [18] studied the absolute worst case ratio of both heuristics and proved that $\omega^{FF}(L)/\omega^*(L) \leqslant 1.75$ and that $\omega^{FFD}(L)/\omega^*(L) \leqslant 1.5$.

Now, we describe the first algorithm for the scheduling problem. We index the independent sets $U_1, U_2, ...,$ initially defined as empty sets and we place the jobs $j \in J$ in the order of a given list $L = (j_1, ..., j_n)$.

ALGORITHM 1(*NF*). To place j_i, take the highest index k such that $U_k \neq \emptyset$. If $U_k \cup \{j_i\}$ remains independent and $c(j_i) + \sum_{j \in U_k} c(j) \leqslant 1$ then place j_i into U_k; otherwise place j_i into the next empty set U_{k+1}.

ALGORITHM 1(*FF*). To place j_i, find the smallest index k such that $U_k \cup \{j_i\}$ is independent and that $c(j_i) + \sum_{j \in U_k} c(j) \leqslant 1$ and place j_i into U_k.

ALGORITHM 1(*FFD*). Arrange the list $L = (j_1, ..., j_n)$ into non-increasing order $e(j_1) \geqslant \cdots \geqslant e(j_n)$ and apply the Algorithm 1(*FF*) on the derived list.

As abbreviations for the algorithms we use Alg 1(*NF*), Alg 1(*FF*) and Alg 1(*FFD*).

THEOREM 1. *There is a problem instance* $(G = (J, E), e)$ *with unit-execution times* $e(j) = 1/k$ *for each job* $j \in J$ (*with* $k = |J|/2$) *and a list* $L = (j_1, ..., j_n)$ *such that*

$$\omega^{Alg\ 1\ (NF)} = \omega^{Alg\ 1\ (FF)} = \omega^{Alg\ 1\ (FFD)} = O(|J|) \cdot \omega^*.$$

Proof. Let J be $\{a_1, ..., a_k\} \cup \{b_1, ..., b_k\}$ and let E be $\{\{a_i, b_j\} \mid 1 \leqslant i \neq j \leqslant k\}$. As list L we use $(a_1, b_1, ..., a_k, b_k)$. Then, Algorithm 1 generates k independent sets $\{a_1, b_1\}, ..., \{a_k, b_k\}$. On the other hand, the optimum solution is given by two independent sets $\{a_1, ..., a_k\}$ and $\{b_1, ..., b_k\}$. Therefore, $\omega^* = 2$ and $\omega^{\text{Alg 1}(NF)} = \omega^{\text{Alg 1}(FF)} = \omega^{\text{Alg 1}(FFD)} = |J|/2$. ∎

3. COLORING METHOD

In this section, we propose a method which generates the constant worst case ratio 2.7 for graphs that can be colored in polynomial time. The algorithm computes in the first step a minimum coloring and applies then a bin-packing heuristic to each color set.

ALGORITHM 2.

Step 1. Compute a minimum partition into independent sets $U_1, ..., U_{\chi(G)}$ for the conflict graph G.

Step 2. Apply a bin-packing heuristic (NF, FF, FFD) to each independent set U_i, $1 \leqslant i \leqslant \chi(G)$.

As above, we denote with Alg 2(NF), Alg 2(FF) and Alg 2(FFD) the corresponding composed algorithms.

THEOREM 2.

$$\omega^{Alg\,2\,(NF)} \leqslant 3 \cdot \omega^*.$$

Proof. Let $U_i^{(1)}, ..., U_i^{(\ell_i)}$ be the independent sets generated by the NF algorithm applied on U_i. Each of these independent sets can be processed on one machine with execution time at most 1. We obtain the following (in)-equalities:

(a) $\omega^{\text{Alg } 2(NF)} = \sum_{i=1}^{\chi(G)} \ell_i$.

(b) $\omega^* \geqslant \chi(G)$.

(c) $\omega^* > \sum_{i=1, \ell_i > 1}^{\chi(G)} \lfloor \ell_i/2 \rfloor$.

The inequality (c) is true, since in the NF algorithm the levels of neighbouring bins B_i, B_{i+1} satisfy: $b_i + b_{i+1} > 1$. We can transform (c) directly into

(c′) $\sum_{i:\, \ell_i > 1} \ell_i < 2\omega^* + \sum_{i:\, \ell_i > 1} 1$.

The (in-) equalities (a), (b) and (c′) imply

$$\begin{aligned} \omega^{\text{Alg } 2(NF)} &= \sum_{i:\, \ell_i = 1} \ell_i + \sum_{i:\, \ell_i > 1} \ell_i \\ &\leqslant 2 \cdot \omega^* + \sum_{i:\, \ell_i \geqslant 1} 1 \\ &= 2 \cdot \omega^* + \chi(G) \leqslant 3 \cdot \omega^*. \quad \blacksquare \end{aligned}$$

We note that there is a set of instances with asymptotically worst case bound 3 for Alg 2(NF). Let ℓ be an even number. As job set we take $J^{(\ell)} = C^{(\ell)} \cup U_1^{(\ell)} \cup U_2^{(\ell)}$ where $C^{(\ell)} = \{c_1, ..., c_{\ell/2}\}$ and $U_i^{(\ell)} = \{u_{i,1}, ..., u_{i,\ell}\}$ for $i = 1, 2$. The edge set in the conflict graph is:

$$E^{(\ell)} = \{\{c, c'\} \mid c, c' \in C^{(\ell)}, c \neq c'\} \cup \{\{c, u_{1,j}\} \mid 1 \leqslant j \leqslant \ell, c \in C^{(\ell)}\}.$$

The execution times of the jobs are: $e(c) = \delta$, $e(u_{1,j}) = 3\delta$ and $e(u_{2,j}) = 1/2 - \delta$ where $\delta = 1/3\ell$.

The chromatic number of the conflict graph is $\ell/2 + 1$ and an optimum solution is given by $\ell/2 + 1$ independent sets:

$$\{u_{2,1}, u_{2,2}, c_1\}, ..., \{u_{2,\ell-1}, u_{2,\ell}, c_{\ell/2}\}, \{u_{1,1}, ..., u_{1,\ell}\}.$$

Another minimum partition into independent sets of G is:

$$\{c_1\}, ..., \{c_{\ell/2}\}, \{u_{2,1}, u_{1,1}, ..., u_{2,\ell}, u_{1,\ell}\}.$$

The bin-packing heuristic NF applied to last set generates ℓ independent sets $\{u_{2,1}, u_{1,1}, ..., \{u_{2,\ell}, u_{1,\ell}\}$, and in total the heuristic Alg 2(NF) generates $\ell/2+\ell$ sets.

Comparing both values, we get

$$\lim_{l \to \infty} \frac{\ell/2+\ell}{\ell/2+1} = 3.$$

THEOREM 3. *If $G=(J, E)$ is a bipartite graph, then $\omega^{Alg\ 2(NF)} \leqslant 2 \cdot \omega^*$.*

Proof. We obtain as above $\omega^{\text{Alg } 2(NF)} = \ell_1 + \ell_2$. For $\ell_1, \ell_2 > 1$ the inequality (c) implies $\omega^* \geqslant \lfloor \ell_1/2 \rfloor + \lfloor \ell_2/2 \rfloor + 1 \geqslant (\ell_1+\ell_2)/2$. In this case $\omega^{\text{Alg } 2(NF)} = \ell_1 + \ell_2 \leqslant 2\omega^*$. For $\ell_1 = \ell_2 = 1$ we have $\omega^{\text{Alg } 2(NF)} = \omega^*$. The remaining case is $\ell_1 = 1$ and $\ell_2 > 1$. Let $b_1, ..., b_{\ell_2}$ be the execution times of the constructed independent sets. Since the bin-packing heuristic NF is applied to the large set U_2, we get the inequalities

$$b_1 + b_2, b_3 + b_4, ..., b_{\ell_2 - 1} + b_{\ell_2} > 1.$$

We obtain that $\omega^* > \lfloor \ell_2/2 \rfloor$. If ℓ_2 is an even number then $\ell_2 < 2\omega^*$ or $\ell_2 \leqslant 2\omega^* - 1$. If ℓ_2 is odd then $\ell_2 < 2\omega^* + 1$. Since $2\omega^* + 1$ is also odd, we have $\ell_2 \leqslant 2\omega^* - 1$. Therefore, $\omega^{\text{Alg } 2(NF)} = 1 + \ell_2 \leqslant 1 + 2\omega^* - 1 \leqslant 2 \cdot \omega^*$. ∎

The bound $2 \cdot \omega^*$ can be achieved even for Alg 2(FFD). As example we take 7 jobs $J = \{j_1, ..., j_7\}$ with one edge $\{j_2, j_7\}$ in the conflict graph. As execution times we choose $e(j_1) = 1/2 - \varepsilon/2$, $e(j_2) = 1/2 - \varepsilon$, $e(j_3) = 1/3 + \varepsilon$, $e(j_4) = 1/3$, $e(j_5) = 1/6$, $e(j_6) = 1/6$, and $e(j_7) = \varepsilon/2$.

An optimum solution is given by two color sets $U = \{j_1, j_4, j_5, j_7\}$ and $U' = \{j_2, j_3, j_6\}$, both with processing times $e(U) = e(U') = 1$. Another possible 2-coloring is given by $\{j_1, j_2, j_3, j_4, j_5, j_6\}$ and $\{j_7\}$. If we apply FFD on these two sets, we obtain four sets $\{j_1, j_2\}$, $\{j_3, j_4, j_5\}$, $\{j_6\}$ and $\{j_7\}$. This gives us the ratio $\omega^{\text{ALG } 2(FFD)}/\omega^* = 2$.

THEOREM 4.

$$\omega^{Alg\ 2(FF)} \leqslant 2.7 \cdot \omega^*.$$

Proof. Let $U_1, ..., U_{\chi(G)}$ be a minimum partition of G into independent sets. We define a weighting function W: $[0, 1] \to [0, 8/5]$ as in [7]:

$$W(\alpha) = \begin{cases} 6/5\alpha & \text{for} \quad 0 \leqslant \alpha \leqslant 1/6, \\ 9/5\alpha - 1/10 & \text{for} \quad 1/6 < \alpha \leqslant 1/3, \\ 6/5\alpha + 1/10 & \text{for} \quad 1/3 < \alpha \leqslant 1/2, \\ 6/5\alpha + 4/10 & \text{for} \quad 1/2 < \alpha \leqslant 1. \end{cases}$$

Furthermore, let $\bar{W}(U)=\sum_{\alpha \in U} W(e(\alpha))$ and $\bar{W}=\sum_{\alpha \in J} W(e(\alpha))$. Using a result about the weighting function in [7] for each color class, the number of bins generated by FF on a set U_i is bounded by $\bar{W}(U_i)+1$. This implies that the total number of bins

$$\omega^{\text{Alg } 2(FF)} \leqslant \sum_{i=1}^{\chi(G)} \bar{W}(U_i)+1$$

is at most $\bar{W}+\chi(G)$. Again, we know from [7] that the weight $\bar{W} \leqslant 1.7 \cdot \omega^*(L)$ where $\omega^*(L)$ is the minimum number of bins in a packing for all jobs without the incompatibility constraint. It is clear that $\omega^*(L) \leqslant \omega^*(G)$. Therefore, $\omega^{\text{Alg } 2(FF)} \leqslant \bar{W}+\chi(G) \leqslant 1.7 \cdot \omega^*(L)+\chi(G) \leqslant 1.7\omega^*(G)+\omega^*(G)=2.7\omega^*(G)$. ∎

The bound 2.7 can be achieved asymptotically for Alg 2(FF) and the worst case ratio for Alg 2(FFD) lies between 2.691 and 2.7. As job set for Alg 2(FF) we take $\{a_{i,k}, b_{i,k} \mid 1 \leqslant i \leqslant 10, 1 \leqslant k \leqslant \ell\} \cup \{c_k, d_k \mid 1 \leqslant k \leqslant 10\ell\}$ and as edge set $E^{(\ell)}=\{\{d_k, d_{k'}\} \mid 1 \leqslant k \neq k' \leqslant 10\ell\}$. The execution times are described in the following table. We assume that $\delta \ll \varepsilon$.

jobs	execution times
$a_{1,k}, ..., a_{5,k}$	$\frac{1}{6}+\frac{\varepsilon}{3^k}-\delta$
$a_{6,k}, ..., a_{10,k}$	$\frac{1}{6}-\frac{\varepsilon}{3^{k+1}}-\delta$
$b_{1,k}, ..., b_{5,k}$	$\frac{1}{3}+\frac{\varepsilon}{3^{k-1}}-\delta$
$b_{6,k}, ..., b_{10,k}$	$\frac{1}{3}-\frac{\varepsilon}{3^k}-\delta$
$c_1, ..., c_{10\ell}$	$\frac{1}{2}+\delta$
$d_1, ..., d_{10\ell}$	δ.

Denote the set of all $a_{i,k}$, $b_{i,k}$ and c_i by A, B and C, respectively. An optimum coloring of the conflict graph is given by the independent sets $A \cup \{d_1\}$, $B \cup \{d_2\}$, $C \cup \{d_3\}$ and $\{d_i\}$ for $4 \leqslant i \leqslant 10\ell$. If we apply FF on the set A with given list

$$(a_{1,1}, a_{2,1}, a_{3,1}, a_{6,1}, a_{7,1}, a_{4,1}, a_{5,1}, a_{8,1}, a_{9,1}, a_{10,1}, a_{1,2}, ...),$$

the bin $B_{2(k-1)+1}$ consists of $\{a_{1,k}, a_{2,k}, a_{3,k}, a_{6,k}, a_{7,k}\}$ and the bin B_{2k} consists of $\{a_{4,k}, a_{5,k}, a_{8,k}, a_{9,k}, a_{10,k}\}$ for $1 \leqslant k \leqslant \ell$. Therefore, FF on $A \cup \{d_1\}$ gives 2ℓ sets.

Applying FF on the list $(b_{1,1}, b_{6,1}, ..., b_{5,1}, b_{10,1}, b_{1,2}, ...)$ generates 5ℓ sets $\{b_{i,k}, b_{i+5,k}\}$. Furthermore, FF on $C \cup \{d_3\}$ generates 10ℓ sets. In total, algorithm Alg 2(FF) gives $2\ell+5\ell+10\ell+10\ell-3=27\ell-3$ sets.

On the other hand, an optimum solution of the scheduling problem has at most $10\ell+2$ sets:

job sets	indices	number of sets
$\{a_{i,k}, b_{5+i,k}, c_{5(k-1)+i}, d_{5(k-1)+i}\}$	$1 \leqslant i \leqslant 5,\ 1 \leqslant k \leqslant \ell$	5ℓ
$\{a_{5+i,k-2}, b_{i,k}, c_{5(\ell+k-3)+i}, d_{5(\ell+k-3)+i}\}$	$1 \leqslant i \leqslant 5,\ 3 \leqslant k \leqslant \ell$	$5\ell-10$
$\{c_{10(\ell-1)+i}, d_{10(\ell-1)+i}, b_{i,1}\}$	$1 \leqslant i \leqslant 5$	5
$\{c_{10(\ell-1)+5+i}, d_{10(\ell-1)+5+i}, b_{i,2}\}$	$1 \leqslant i \leqslant 5$	5
$\{a_{6,\ell-1}, ..., a_{10,\ell-1}\}$		1
$\{a_{6,\ell}, ..., a_{10,\ell}\}$		1

Comparing both values, we obtain

$$\lim_{\ell \to \infty} \frac{27\ell-3}{10\ell+2} = 2.7$$

For Alg 2(*FFD*) we construct the instance as follows. We take k series ($k \ll \ell$) of ℓ jobs $a_{i,1}, ..., a_{i,\ell}$ with execution times

$$e(a_{i,1}) = \cdots = e(a_{i,\ell}) = \frac{1}{(b_i+1)} + \varepsilon$$

where the sequence b_i is defined recursively:

$$b_1 = 1$$

$$b_i = b_{i-1} \cdot (b_{i-1}+1) \qquad \text{for} \quad i \geqslant 2.$$

Furthermore, we take ℓ jobs $a_1, ..., a_\ell$ with execution times $e(a_i) = \varepsilon$. The edge set $E^{(\ell)}$ is equal to $\{\{a_i, a_j\} \mid 1 \leqslant i < j \leqslant \ell\}$. We take an $\varepsilon < 1/((k+1)\, b_{k+1})$. The first values of the execution times above are $\frac{1}{2}+\varepsilon$, $\frac{1}{3}+\varepsilon$, $\frac{1}{7}+\varepsilon$, $1/(6 \cdot 7+1)+\varepsilon$, ... Notice, that the sum of the first k execution times $\sum_{j=1}^{k} (b_j+1)^{-1}$ is equal to $1-1/(b_k+1)$.

An optimum solution is given by ℓ independent sets $U_i = \{a_i, a_{1,i}, ..., a_{k,i}\}$ ($1 \leqslant i \leqslant \ell$) with total processing time $e(U_i) = (k+1)\varepsilon + \sum_{j=1}^{k} (b_j+1)^{-1} = (k+1)\varepsilon + 1 - 1/b_{k+1} < 1$.

On the other hand, a possible coloring of the conflict graph is also given by k independent sets $U_i = \{a_i, a_{i,1}, ..., a_{i,\ell}\}$ ($1 \leqslant i \leqslant k$) and $\ell-k$ independent sets $U_i = \{a_i\}$ ($k+1 \leqslant i \leqslant \ell$). If we apply *FFD* on a set U_i of the first type, we get $\lceil \ell/b_i \rceil$ sets. In total, the algorithm Alg 2(*FFD*) generates

$$\sum_{i=1}^{k} \left\lceil \frac{\ell}{b_i} \right\rceil + (\ell-k)$$

independent sets. Using $k \ll \ell$ and $\lim_{k \to \infty} \sum_{i=1}^{k} 1/b_i = 1.691...$, the quotient $\omega^{\text{Alg } 2(FFD)}/\omega^*$ tends to 2.691... for $k, \ell \to \infty$.

4. PRECOLORING METHOD

In this section, we analyse an approximation method where the "long" jobs $\bar{J} = \{ j \in J \mid e(j) > \frac{1}{2} \}$ are colored differently. We denote with $\chi_{\bar{J}}(G)$ the minimum number of colors in a coloring with this property. This problem is known in the literature as the precoloring extension problem $1 - PrExt$. Given an undirected graph $G = (V, E)$ and k different vertices $v_1, ..., v_k \in V$, the problem is to find a minimum coloring f of G such that $f(v_i) = i$ for $1 \leqslant i \leqslant k$. This problem is solvable in polynomial time for interval graphs, forests, split graphs, complements of bipartite graphs, cographs, partial K-trees and complements of Meyniel graphs [2, 3, 10, 11, 13, 15]. For the definitions of these graph classes we refer to [5, 9]. On the other hand, $1 - PrExt$ is NP-complete for bipartite graphs [3]. Unknown is the computational complexity of $1 - PrExt$ e.g. for chordal graphs [11].

ALGORITHM 3.

Step 1. Let $\bar{J}$ be the jobs $j \in J$ with execution time $e(j) > \frac{1}{2}$.

Step 2. Compute a partition into independent sets $U_1, ..., U_{\chi_{\bar{J}}(G)}$ for the conflict graph G where $|U_i \cap \bar{J}| \leqslant 1$ for each $1 \leqslant i \leqslant \chi_{\bar{J}}(G)$.

Step 3. Apply a bin-packing heuristic (NF, FF, FFD) to each independent set U_i, $1 \leqslant i \leqslant \chi_{\bar{J}}(G)$.

As above, we denote with Alg 3(NF), Alg 3(FF) and Alg 3(FFD) the corresponding composed algorithms. We can apply these algorithms for graph classes, where the graph problem $1 - PrExt$ is solvable in polynomial time. We notice that the set of instances in Theorem 2 generates the ratio 3 for Alg 3(NF).

LEMMA 5. *Let L be a list of positive numbers $a_1, ..., a_n \leqslant 1$ with $a_\ell > \frac{1}{2}$ for at most one index ℓ, let $B_1, ..., B_m$ (with $m > 2$) be the bins generated by FF on L. If b_i is the level of bin B_i for $1 \leqslant i \leqslant m$, then*

$$\sum_{i=1}^{m} b_i \geqslant \tfrac{2}{3} \cdot m - \tfrac{1}{2}.$$

Proof. Let B_k be the bin which contains a_ℓ. If all levels $b_i \geqslant \frac{2}{3}$ for $1 \leqslant i < m$, then the sum

$$\sum_{i=1}^{m} b_i \geqslant \tfrac{2}{3}(m-2) + b_{m-1} + b_m > \tfrac{2}{3}(m-2) + 1 > \tfrac{2}{3}m - \tfrac{1}{2}.$$

Let us assume that there is an index $j < m$ such that $b_j < \frac{2}{3}$. We choose the smallest index j with this property. It follows that $b_{j'} \geqslant \frac{2}{3}$ for each $j' < j$. Since $b_j < \frac{2}{3}$, each number x in the bins $B_{j+1}, ..., B_m$ is larger than $\frac{1}{3}$. If $m = k$ we define $m' = m - 1$ (otherwise we define $m' = m$). Since $a_i \leqslant \frac{1}{2}$ for $i \neq \ell$, the bins $B_{j+1}, ..., B_{m'-1}$ (with exception of B_k) contain at least two integers and, therefore, the levels $b_{j+1}, ..., b_{m'-1}$ (with exeception of b_k) are larger than $\frac{2}{3}$.

If $k \leqslant j$, then $b_j + b_m > 1$ and $\sum_{i=1}^{m} b_i > \frac{2}{3}(m-2) + 1$. If $k = m$, the bin levels $b_1, ..., b_{j-1}, b_{j+1}, ..., b_{m-2} \geqslant \frac{2}{3}$ and $b_j + b_{m-1} + b_m > \frac{3}{2}$. In this case, the sum

$\sum_{i=1}^{m} b_i \geqslant \frac{2}{3}(m-3) + \frac{3}{2} = \frac{2}{3}m - \frac{1}{2}$. In the remaining case with $j<k<m$, we have $b_j + b_k + b_m > \frac{3}{2}$ and the sum $\sum_{i=1}^{m} b_i$ is again larger than $\frac{2}{3}m - \frac{1}{2}$. ▮

THEOREM 6.

$$\omega^{Alg\ 3(FF)} \leqslant 2.5 \cdot \omega^*.$$

Proof. We consider a partition into independent sets $U_1, ..., U_{\chi_{\bar{J}}(G)}$ where $\chi_{\bar{J}}(G)$ is the minimum number of colors in a coloring such that the precoloring for $\bar{J}$ is fulfilled. We denote with k_i the number of bins generated by FF on U_i. Now, we define

$$a = |\{i \mid 1 \leqslant i \leqslant \chi_{\bar{J}}(G), k_i = 1, U_i \cap \bar{J} \neq \emptyset\}|,$$

$$b = |\{i \mid 1 \leqslant i \leqslant \chi_{\bar{J}}(G), k_i = 1, U_i \cap \bar{J} = \emptyset\}|.$$

Lemma 5 implies that the following inequality is true:

$$\omega^* \geqslant \frac{a}{2} + \sum_{i:\,k_i=2} 1 + \sum_{i:\,k_i>2} \left\lceil \frac{2}{3} \cdot k_i - \frac{1}{2} \right\rceil.$$

This inequality can be transformed into

$$\sum_{i:\,k_i>2} k_i \leqslant \tfrac{3}{2}\omega^* - \tfrac{3}{4}a - \sum_{i:\,k_i=2} \tfrac{3}{2} + \sum_{i:\,k_i>2} \tfrac{3}{4}.$$

Algorithm 3 composed with FF generates

$$\begin{aligned}\omega^{\text{Alg } 3(FF)} &= a + b + \sum_{i:\,k_i=2} 2 + \sum_{i:\,k_i>2} k_i \\ &\leqslant \tfrac{3}{2}\omega^* + [\tfrac{1}{4}]a + b + \sum_{i:\,k_i=2} \tfrac{1}{2} + \sum_{i:\,k_i>2} \tfrac{3}{4} \\ &\leqslant \tfrac{3}{2}\omega^* + a + b + \sum_{i:\,k_i=2} 1 + \sum_{i:\,k_i>2} 1.\end{aligned}$$

Since $a + b + \sum_{i:\,k_i>1} 1 = \chi_{\bar{J}}(G)$ and $\chi_{\bar{J}}(G) \leqslant \omega^*$, the last four terms together can be bounded by ω^*. This generates the upper bound 2.5. ▮

COROLLARY 7. *The algorithm Alg* 3(*FF*) *runs in polynomial time and has the absolute worst case ratio* 2.5 *on the following graph classes*: *cographs*, *interval graphs*, *partial K-trees for constant K*, *split graphs*, *complements of bipartite graphs and complements of Meyniel graphs.*

The bound 2.5 can be achieved asymptotically for Alg 3(FF) and the worst case ratio for Alg 3(FFD) lies between 2.423 and 2.5. As job set for Alg 3(FF) we take $\{a_{i,k} \mid 1 \leqslant i \leqslant 2l,\ \ 1 \leqslant k \leqslant \ell\} \cup \{b_k \mid 1 \leqslant k \leqslant \ell^2\} \cup \{c_k \mid 1 \leqslant k \leqslant \ell^2 + \ell + 1\}$ and as edge set $E^{(\ell)} = \{\{c_k, c_{k'}\} \mid 1 \leqslant k \neq k' \leqslant \ell^2 + \ell + 1\}$. The execution times are described in the following table. We assume that $\delta \ll \varepsilon$.

jobs	execution times
$a_{1,k}, \ldots, a_{\ell,k}$	$\frac{1}{3} - 3^{k-1} \cdot \varepsilon - \delta$
$a_{\ell+1,k}, \ldots, a_{2\ell,k}$	$\frac{1}{3} + 3^{k} \cdot \varepsilon - \delta$
$b_1, \ldots, b_{\ell^2}$	$\frac{1}{3} + \delta$
$c_1, \ldots, c_{\ell^2+\ell+1}$	$\delta.$

Denote the set of all $a_{i,k}$ and b_k by A and B, respectively. An optimum coloring of the conflict graph is given by ℓ independent sets $\{a_{i,k} \mid 1 \leqslant i \leqslant 2\ell\} \cup \{c_k\}$ (for $1 \leqslant k \leqslant l$), by one set $B \cup \{c_{\ell+1}\}$ and ℓ^2 sets $\{c_k\}$ for $\ell+2 \leqslant k \leqslant \ell^2+\ell+1$. *FF* on the list $(a_{1,k}, a_{\ell+1,k}, \ldots, a_{\ell,k}, a_{2\ell,k})$ generates ℓ sets $\{a_{i,k}, a_{\ell+i,k}\}$. Therefore, *FF* on the first ℓ sets produces ℓ^2 sets. Furthermore, *FF* on $B \cup \{c_{\ell+1}\}$ generates $\lceil \ell^2/2 \rceil$ sets. In total, algorithm Alg 3(*FF*) produces $2\ell^2 + \lceil \ell^2/2 \rceil = \lceil \frac{5}{2}\ell^2 \rceil$ sets.

On the other hand, an optimum solution of the scheduling problem has only $\ell^2+\ell+1$ sets:

job sets	indices	number of sets
$\{a_{\ell+i,k}, a_{i,k+1}, b_{\ell(k-1)+i}, c_{\ell(k-1)+i}\}$	$1 \leqslant i \leqslant \ell,\ 1 \leqslant k < l$	$\ell(\ell-1)$
$\{a_{i,1}, a_{\ell+1,\ell}, c_{\ell(\ell-1)+i}\}$	$1 \leqslant i \leqslant \ell$	ℓ
$\{b_{\ell(\ell-1)+i}, c_{\ell^2+i}\}$	$1 \leqslant i \leqslant \ell$	ℓ
$\{c_{\ell^2+\ell+1}\}$		1

Comparing both values, we obtain

$$\lim_{\ell \to \infty} \frac{\lceil \frac{5}{2}\ell^2 \rceil}{\ell^2+\ell+1} = 2.5.$$

We construct the instance for Alg 3(*FFD*) as follows. First, we take one series of 2ℓ jobs $a_{1,.1}, \ldots, a_{1,2\ell}$ with execution times

$$e(a_{1,1}) = \cdots = e(a_{1,2\ell}) = \tfrac{1}{3} + \varepsilon.$$

Next, we use k series $(k \ll \ell)$ of ℓ jobs $a_{i,1}, \ldots, a_{i,\ell}$ $(2 \leqslant i \leqslant k+1)$ with execution times

$$e(a_{i,1}) = \cdots = e(a_{i,\ell}) = \frac{1}{(b_i+1)} + \varepsilon$$

where the sequence b_i is defined recursively:

$$b_2 = 3 \qquad \text{for} \quad i = 2.$$

$$b_i = b_{i-1} \cdot (b_{i-1}+1) \qquad \text{for} \quad i > 2.$$

Furthermore, we take ℓ jobs $a_1, ..., a_\ell$ with execution times $e(a_i) = \varepsilon$ and the edge set $\{\{a_i, a_j\} \mid 1 \leqslant i < j \leqslant \ell\}$. Here, we use $\varepsilon < 1/((k+3)\, b_{k+2})$. The first values of the execution times $a_{i,k}$ are $\frac{1}{3}+\varepsilon$, $\frac{1}{4}+\varepsilon$, $\frac{1}{13}+\varepsilon$, $1/(12 \cdot 13+1)+\varepsilon$, ... The sum of the values $\sum_{j=2}^{k+1} (b_j+1)^{-1} + \frac{2}{3}$ is equal to $1 - 1/(b_{k+2})$.

An optimum solution is given by ℓ independent sets $U_i = \{a_i, a_{1,i}, a_{1,\ell+i}, a_{2,i}, ..., a_{k+1,i}\}$ with $e(U_i) = (k+3)\varepsilon + \frac{2}{3} + \sum_{j=2}^{k+1} (b_j+1)^{-1} = (k+3)\varepsilon + 1 - 1/(b_{k+2}) < 1$.

On the other hand, a coloring is also given by one independent set $U_1 = (a_1, a_{1,1}, ..., a_{1,2\ell}\}$, k independent sets $U_i = \{a_i, a_{i,1}, ..., a_{i,\ell}\}$ $(2 \leqslant i \leqslant k+1)$ and $\ell - k - 1$ independent sets $U_i = \{a_i\}$ $(k+2 \leqslant i \leqslant \ell)$. If we apply *FFD* on U_1, we generate ℓ independent sets. If we apply *FFD* on U_i $(2 \leqslant i \leqslant k+1)$, we get $\lceil \ell/b_i \rceil$ sets. In total, the algorithm Alg 3(*FFD*) generates at least

$$(\ell - k - 1) + \ell + \sum_{i=2}^{k+1} \left\lceil \frac{\ell}{b_i} \right\rceil$$

independent sets. Using $k \ll \ell$ and $\lim_{k \to \infty} \sum_{i=2}^{k+1} (1/b_i) = 0.423...$, the quotient $\omega^{\text{Alg } 3(FFD)}/\omega^*$ tends to 2.423... for $k, \ell \to \infty$.

5. GENERAL COLORING METHODS

In this section, we study three approximation methods for the scheduling problem where the worst case ratio becomes $2 + 1/k$ with different constants $k = 3$, 5 and 7. The methods can be applied to cographs and partial K-trees.

In the first algorithm we consider the job set $\bar{J} = \{j \in J \mid e(j) > 1/k\}$ where $k > 1$ is a positive integer. Notice, that an independent set U in a solution of the scheduling problem can have at most $k-1$ jobs of $\bar{J}$. Therefore, we compute in the first step of the algorithm a coloring of the conflict graph where at most $k-1$ jobs lie in one color set. Then, in the second step we apply as in the other algorithms a bin-packing heuristic.

ALGORITHM 4.

Step 1. Let $\bar{J}$ be the jobs $j \in J$ with execution time $e(j) > 1/k$ (where $k \in \mathbb{N}$ and $k > 1$).

Step 2. Compute a minimum coloring with independent sets $U_1, ..., U_m(G)$ for the conflict graph G where $|U_i \cap \bar{J}| \leqslant k-1$, $1 \leqslant i \leqslant m$.

Step 3. Apply a bin-packing heuristic (*NF*, *FF*, *FFD*) to each independent set U_i, $1 \leqslant i \leqslant m$.

LEMMA 8. *Let L be a list of positive numbers $a_1, ..., a_n \leqslant 1$ with at most two numbers $a_\ell, a_{\ell'} > \frac{1}{3}$, let $B_1, ..., B_m$ (with $m > 2$) be the bins generated by FF on L. If b_i is the level of bin B_i for $1 \leqslant i \leqslant m$, then*

$$\sum_{i=1}^{m} b_i \geqslant \tfrac{3}{4} \cdot m - \tfrac{3}{4}.$$

Proof. For $m=3$ using the *FF*-property, we have $b_1+b_2>1$, $b_2+b_3>1$ and $b_1+b_3>1$. This implies $b_1+b_2+b_3>\frac{3}{2}=\frac{3}{4}\cdot 2$.

Now, we assume that $m>3$. Let B_k, $B_{k'}$ be the bins which contain a_ℓ, $a_{\ell'}$, respectively. We define m' as the maximum index of a bin unequal B_k and $B_{k'}$. Notice that $m'\geqslant m-2$. If all levels $b_i\geqslant\frac{3}{4}$ for $1\leqslant i<m'$, then the sum

$$\sum_{i=1}^{m} b_i\geqslant\tfrac{3}{4}(m-3)+\tfrac{3}{2}=\tfrac{3}{4}m-\tfrac{3}{4}.$$

Otherwise, we assume that there is an index $j<m'$ such that $b_j<\frac{3}{4}$. We choose the smallest index j with this property. It follows that $b_{j'}\geqslant\frac{3}{4}$ for each $j'<j$. Since $b_j<\frac{3}{4}$, each number x in the bins $B_{j+1}, ..., B_m$ is larger than $\frac{1}{4}$. Since $a_i\leqslant\frac{1}{3}$ for $i\neq\ell,\ell'$, the bins $B_{j+1}, ..., B_{m'-1}$ (with exeception of B_k, $B_{k'}$) are filled more than $\frac{3}{4}$.

By case analysis, we show that $\sum_{i=1}^{m} b_i\geqslant\frac{3}{4}(m-1)$.

$k\leqslant j$, $k=k'$ **or** *$k'\leqslant j$.* Here, we have at most three critical bins and obtain $\sum_{i=1}^{m} b_i\geqslant\frac{3}{4}(m-3)+\frac{3}{2}$.

$k=m$ **or** *$k'=m$.* By symmetry and the first case, we consider only $k=m$ and $j<k'<m$. If $k'<m'$ then $b_j, b_{k'}\geqslant\frac{2}{3}$ and $b_{m'}+b_m>1$. Now, let $k'>m'$. The bin $B_{m'}$ can have one, two or three numbers and each number $y\in B_{m'}$ lies in the interval $(\frac{1}{4},\frac{1}{3}]$. In the case that $B_{m'}$ has only one element $b_j, b_{k'}, b_k>\frac{2}{3}$ and $b_j+b_{m'}+b_{k'}+b_m>\frac{4}{3}+1$. If $B_{m'}$ has at least two elements, then $b_j+b_{m'}>1+\frac{1}{4}$ and $b_{k'}+b_m>1$. In all these cases, the sum of the levels gives at least $\frac{3}{4}(m-4)+\frac{9}{4}=\frac{3}{4}(m-1)$.

$j<k<k'<m$. In this case m' is equal to m. Let y be a number in B_m. Again, we know that $y\in(\frac{1}{4},\frac{1}{3}]$. This implies that $b_k, b_{k'}>\frac{2}{3}$, and, therefore, $b_j+b_k+b_{k'}+b_m>\frac{7}{3}$. In total, $\sum_{i=1}^{m} b_i\geqslant\frac{3}{4}(m-4)+\frac{7}{3}>\frac{3}{4}(m-1)$. ∎

THEOREM 9. *For $k=3$,*

$$\omega^{Alg\,4(FF)}\leqslant 2.\overline{3}\cdot\omega^*.$$

Proof. Here, Lemma 8 implies the following inequality:

$$\omega^*\geqslant\frac{a}{2}+\sum_{k_i=2}1+\sum_{k_i\geqslant 3}\left\lceil\frac{3}{4}k_i-\frac{3}{4}\right\rceil.$$

Using the fact that the number of independent set $m\leqslant\omega^*$, we have

$$\begin{aligned}\omega^{\mathrm{Alg}\,4(FF)}&=a+b+\sum_{k_i=2}2+\sum_{k_i\geqslant 3}k_i\\&\leqslant\tfrac{4}{3}\omega^*+b+\tfrac{1}{3}a+\sum_{k_i=2}\tfrac{2}{3}+\sum_{k_i\geqslant 3}1\\&\leqslant\tfrac{4}{3}\omega^*+m\leqslant\tfrac{7}{3}\omega^*.\quad ∎\end{aligned}$$

The bound $2.\overline{3}$ can be achieved for $k\geqslant 4$ and Alg 4(*FF*). This implies that the worst case ratio $2.\overline{3}$ cannot be improved using larger numbers $k\geqslant 4$. To prove this for $k=4$ we construct the conflict graph $G^{(\ell)}=(J^{(\ell)}, E^{(\ell)})$ as follows. As job set $J^{(\ell)}$ we take the union of $\{a_{i,k} \mid 1 \leqslant i \leqslant 4,\ 1 \leqslant k \leqslant 2\ell\}$ and $\{b_k \mid 1 \leqslant k \leqslant 3\ell\}$ and as edge set $E^{(\ell)}=\{\{b_k, b_{k'}\} \mid 1 \leqslant k < k' \leqslant 3\ell\}$. The execution times of the jobs are

jobs	execution times
$a_{1,k}$	$\frac{1}{2}-\varepsilon-\delta$
$a_{2,k}$	$2\varepsilon-\delta$
$a_{3,k}$	$\frac{1}{2}+\frac{\varepsilon}{2^{k-1}}-\delta$
$a_{4,k}$	$\frac{1}{2}-\frac{\varepsilon}{2^k}-\delta$
$b_1, ..., b_{3\ell}$	$\delta \ll \varepsilon$

An optimum coloring where each color set contains at most three jobs larger than $\frac{1}{4}$ is given by independent sets $U_k=\{a_{i,k} \mid 1 \leqslant i \leqslant 4\} \cup \{b_k\}$ for $1 \leqslant k \leqslant 2l$ and $U_k=\{b_k\}$ for $2\ell+1 \leqslant k \leqslant 3\ell$. Notice that $G^{(\ell)}$ contains a clique of size 3ℓ. If we apply *FF* on these color sets, we get $6\ell+\ell=7\ell$ sets.

On the other hand, an optimum solution for the scheduling problem has at most $3\ell+1$ sets:

job sets	indices	number of sets
$\{a_{4,k}, a_{3,k+1}, b_k\}$	$1 \leqslant k < 2\ell$	$2\ell-1$
$\{a_{1,k}, a_{1,l+k}, b_{2l-1+k}\}$	$1 \leqslant k \leqslant l$	ℓ
$\{a_{3,1}, a_{2,1}, ..., a_{2,2\ell}, b_{3\ell}\}$		1
$\{a_{4,2\ell}\}$		1

In total, $\lim_{\ell \to \infty}(7\ell/(3\ell+1))=7/3$.

In the second method, we allow that a color set has at most ℓ jobs with execution time in the interval $(1/(\ell+1), 1/\ell]$ for each $1 \leqslant \ell \leqslant k$. We have to compute an optimum coloring such that each color set satisfies this property. In the second step of the approximation algorithm we use again a bin-packing heuristic.

ALGORITHM 5.

Step 1. Let $\bar{J}_\ell$ be the jobs $j \in J$ with execution time $e(j) \in (1/(\ell+1), 1/\ell]$ (for each $1 \leqslant \ell \leqslant k$).

Step 2. Compute a minimum coloring with independent sets $U_1, ..., U_m(G)$ for the conflict graph G where $|U_i \cap \bar{J}_\ell| \leqslant \ell$ for each $1 \leqslant \ell \leqslant k$ and $1 \leqslant i \leqslant m$.

Step 3. Sort the jobs with execution time $e(j) > 1/(k+1)$ in each independent set U_i, $1 \leqslant i \leqslant m$.

Step 4. Apply a bin-packing heuristic (*NF*, *FF*, *FFD*) to each independent set U_i, $1 \leqslant i \leqslant m$.

LEMMA 10. *Let k be a positive integer and let L be a list of positive numbers $1 \geqslant a_1 \geqslant \cdots \geqslant a_{n'} > 1/(k+1) \leqslant a_{n'+1}, ..., a_n$ where $|\{ j \mid a_j \in (1/(\ell+1), 1/\ell] \}| \leqslant \ell$ for $1 \leqslant \ell \leqslant k$. If $B_1, ..., B_m$ are the bins generated by FF on L and if b_i is the level of bin B_i, $1 \leqslant i \leqslant m$, then for $m \geqslant k+1$:*

$$\sum_{i=1}^{m} b_i \geqslant \frac{k+1}{k+2} \cdot (m-k-1) + \frac{k(k-1)}{k+1} + 1.$$

Moreover, for $m = k \geqslant 2$ we get:

$$\sum_{i=1}^{k} b_i \geqslant \frac{(k-2)(k-1)}{k} + 1.$$

Proof. For $m = k+1$, the last bin B_{k+1} contains at least one number $y \leqslant 1/(k+1)$—the other numbers larger than $1/(k+1)$ can be placed into the first k bins. Then, $b_1, ..., b_k \geqslant k/(k+1)$ and

$$\sum_{i=1}^{k+1} b_i = \sum_{i=1}^{k-1} b_i + b_k + b_{k+1} \geqslant \frac{(k-1)k}{k+1} + 1.$$

Therefore, the inequality is true for $m = k+1$ and also for $m = k$.

Let $m > k+1$. If all levels $b_{k+1}, ..., b_{m-1} \geqslant (k+1)/(k+2)$, then the assertion follows directly. We assume that there is an index $k+1 \leqslant j < m$ such that $b_j < (k+1)/(k+2)$. Choosing the smallest index j with this property, $b_{j'} \geqslant (k+1)/(k+2)$ for $k+1 \leqslant j' < j$. Since $b_j < (k+1)/(k+2)$, each number x in the further bins $B_{j+1}, ..., B_{m-1}$ is larger than $1/(k+2)$ and each of these bins contains at least $k+1$ numbers (note that each of these numbers lies in the interval $(1/(k+2), 1/(k+1)]$). Since B_j contains at least $k+1$ numbers and $b_j < (k+1)/(k+2)$, there is at least one number $y \in B_j$ with $y < 1/(k+2)$. From this reason we know that $b_1, ..., b_k > (k+1)/(k+2)$. In total, we have

$$b_1, ..., b_{j-1}, b_{j+1}, ..., b_{m-1} \geqslant \frac{k+1}{k+2},$$

$$b_j + b_m > 1.$$

Therefore, the sum of the levels can be bounded as follows:

$$\begin{aligned}\sum_{i=1}^{m} b_i &> \frac{k+1}{k+2}(m-2) + 1 = \frac{k+1}{k+2}(m-k-1) + \frac{k+1}{k+2}(k-1) + 1 \\ &\geqslant \frac{k+1}{k+2}(m-k-1) + \frac{(k-1)k}{k+1} + 1. \quad \blacksquare\end{aligned}$$

The right sides of the inequalities above have the following values for $k = 1, 2, 3$ and 4. These values will be used later in the proof of the next theorem.

$$k=1: \tfrac{2}{3}(m-2)+1=\tfrac{2}{3}m-\tfrac{1}{3}$$
$$k=2: \tfrac{3}{4}(m-3)+\tfrac{5}{3}=\tfrac{3}{4}m-\tfrac{7}{12}$$
$$k=3: \tfrac{4}{5}(m-4)+\tfrac{5}{2}=\tfrac{4}{5}m-\tfrac{7}{10}$$
$$k=4: \tfrac{5}{6}(m-5)+\tfrac{17}{5}=\tfrac{5}{6}m-\tfrac{23}{30}$$

THEOREM 11. *For $k=1, 2, 3, 4$*

$$\omega^{Alg\ 5(FF)} \leqslant \left(2+\frac{1}{k+1}\right)\cdot\omega^*.$$

Proof. The proof works similar as Theorem 6. Here, Lemma 10 implies the following inequalities for $k=1, 2, 3, 4$:

$$1:\omega^* \geqslant \frac{a}{2}+\sum_{k_i=2} 1+\sum_{k_i\geqslant 3}\left[\frac{2}{3}k_i-\frac{1}{3}\right]$$
$$2:\omega^* \geqslant \frac{a}{2}+\sum_{k_i=2} 1+\sum_{k_i\geqslant 3}\left[\frac{3}{4}k_i-\frac{7}{12}\right]$$
$$3:\omega^* \geqslant \frac{a}{2}+\sum_{k_i=2} 1+\sum_{k_i=3}\frac{5}{3}+\sum_{k_i\geqslant 4}\left[\frac{4}{5}k_i-\frac{7}{10}\right]$$
$$4:\omega^* \geqslant \frac{a}{2}+\sum_{k_i=2} 1+\sum_{k_i=3}\frac{5}{3}+\sum_{k_i=4}\frac{5}{2}+\sum_{k_i\geqslant 5}\left[\frac{5}{6}k_i-\frac{23}{30}\right].$$

With the same method as in Theorem 6, we get as bound for $\omega^{\mathrm{Alg}\ 5(FF)} \leqslant a+b+\sum_{k_i>1} k_i$ the values 2.5, $2.\bar{3}$, 2.25, and 2.2 for $k=1, 2, 3$ and 4, respectively. ∎

The bound $2+\frac{1}{5}$ can be achieved for Alg 5(FFD) and $k\geqslant 4$. This implies that the worst case ratio $\frac{11}{5}$ of Alg 5(FFD) cannot be improved using larger numbers $k\geqslant 5$. To prove this for $k=4$ we construct the conflict graph $G^{(\ell)}=(J^{(\ell)}, E^{(\ell)})$ as follows. As job set $J^{(\ell)}$ we take the union of $\{a_{i,k} \mid 1\leqslant i\leqslant 7, 1\leqslant k\leqslant 2\ell\}$ and $\{b_k \mid 1\leqslant k\leqslant 5\ell\}$ and as edge set $E^{(\ell)}=\{\{b_k, b_{k'}\} \mid 1\leqslant k<k'\leqslant 5\ell\}$. The execution times of the jobs are

jobs	execution times
$a_{1,k}$	$\frac{3}{4}+4^{k-1}\varepsilon-\delta$
$a_{2,k}$	$\frac{3}{8}+4^{k-1}\varepsilon/2-\delta$
$a_{3,k}$	$\frac{3}{8}+4^{k-1}\varepsilon/2-\delta$
$a_{4,k}$	$\frac{1}{4}+4^{k-1}\varepsilon-\delta$
$a_{5,k}$	$\frac{1}{4}-4^{k-1}\varepsilon/4-\delta$
$a_{6,k}$	$\frac{1}{4}-4^{k-1}\varepsilon/4-\delta$
$a_{7,k}$	$\frac{1}{4}-4^{k-1}\varepsilon/4-\delta$
b_k	$\delta \ll \varepsilon$

An optimum coloring where each color set contains at most ℓ jobs $j \in J$ with execution times $e(j) \in (1/(\ell+1), 1/\ell]$ for $1 \leqslant \ell \leqslant 4$ is given by independent sets $U_k = \{a_{i,k} \mid 1 \leqslant i \leqslant 7\} \cup \{b_k\}$ for $1 \leqslant k \leqslant 2\ell$ and $U_k = \{b_k\}$ for $2\ell+1 \leqslant k \leqslant 5\ell$. Notice that $G^{(\ell)}$ contains a clique of size 5ℓ. If we apply *FFD* on these color sets, we get $8\ell + 3\ell = 11\ell$ sets.

On the other hand, an optimum solution for the scheduling problem has $5\ell+1$ sets:

job sets	indices	number of sets
$\{a_{1,k}, a_{5,k+1}, b_k\}$	$1 \leqslant k \leqslant 2\ell - 1$	$2\ell - 1$
$\{a_{2,k}, a_{3,k}, a_{6,k+1}, b_{2\ell-1+k}\}$	$1 \leqslant k \leqslant 2\ell - 1$	$2\ell - 1$
$\{a_{4,k}, a_{7,k+1}, a_{4,\ell+k}, a_{7,\ell+k+1}, b_{4\ell-2+k}\}$	$1 \leqslant k \leqslant \ell - 1$	$\ell - 1$
$\{a_{4,\ell}, a_{7,\ell+1}, a_{5,1}, a_{6,1}, b_{5\ell-2}\}$		1
$\{a_{1,2\ell}, b_{5\ell-1}\}$		1
$\{a_{2,2\ell}, a_{3,2\ell}, b_{5\ell}\}$		1
$\{a_{7,1}, a_{4,2\ell}\}$		1

In total, $\lim_{\ell \to \infty}(11\ell/(5\ell+1)) = 2.2$.

Algorithm 6.

Step 1. Let $\bar{J}_\ell$ be the jobs $j \in J$ with execution time $e(j) \in (1/(\ell+1), 1]$ (for each $1 \leqslant \ell \leqslant k$).

Step 2. Compute a minimum coloring with independent sets $U_1, ..., U_m(G)$ for the conflict graph G where $|U_i \cap \bar{J}_\ell| \leqslant \ell$ for each $1 \leqslant \ell \leqslant k$ and $1 \leqslant i \leqslant m$.

Step 3. Sort the jobs with execution time $e(j) > 1/(k+1)$ in each independent set U_i, $1 \leqslant i \leqslant m$.

Step 4. Apply a bin-packing heuristic (*NF*, *FF*, *FFD*) to each independent set U_i, $1 \leqslant i \leqslant m$.

Lemma 12. *Let k be a positive integer and let L be a list of positive numbers $1 \geqslant a_1 \geqslant \cdots \geqslant a_n > 1/(k+1)$ where $|\{j \mid 1 \leqslant j \leqslant n, a_j \in (1/(\ell+1), 1]\}| \leqslant \ell$ for $1 \leqslant \ell \leqslant k$. If $B_1, ..., B_m$ is the set of non-empty bins generated by FF on L and if b_i is the level of bin B_i, $1 \leqslant i \leqslant m$, then*

$$\sum_{i=1}^{m} b_i \geqslant \begin{cases} 1 & \text{for} \quad m=2 \\ \frac{7}{4} & \text{for} \quad m=3 \\ \frac{31}{11} & \text{for} \quad m=4. \end{cases}$$

Proof. $m=2$. We get directly $b_1 + b_2 > 1$.

$m=3$. Let y be a number in the last bin B_3. Since we have at most three numbers in $(\frac{1}{4}, 1]$, at most two in $(\frac{1}{3}, 1]$ and at most one in $(\frac{1}{2}, 1]$, we get that $y \leqslant \frac{1}{4}$. Therefore, $b_1, b_2 \geqslant \frac{3}{4}$ and $b_1 + b_2 + b_3 \geqslant \frac{3}{4} + 1$.

$m=4$. Again, let y be a number in the last bin B_4. Here, we can show that $y \leqslant \frac{1}{11}$—since the other larger numbers are placed into the first three bins. Then, using $y \leqslant \frac{1}{11}$, we obtain $b_1, b_2, b_3 \geqslant \frac{10}{11}$ and $b_1+b_2+b_3+b_4 \geqslant \frac{20}{11}+1$. ∎

LEMMA 13. *Let k be a positive integer $3 \leqslant k \leqslant 10$, let L be a list of positive numbers $1 \geqslant a_1 \geqslant \ldots \geqslant a_{n'} > 1/(k+1) \leqslant a_{n'+1}, \ldots, a_n$ where $|\{j \mid a_j \in (1/(\ell+1), 1]\}| \leqslant \ell$ for $1 \leqslant \ell \leqslant k$. If $B_1, \ldots, B_m$ is the set of bins generated by FF on L and if b_i is the level of bin B_i, $1 \leqslant i \leqslant m$, then for $m \geqslant 4$:*

$$\sum_{i=1}^{m} b_i \geqslant \frac{k+1}{k+2} \cdot (m-4) + \frac{2k}{k+1} + 1.$$

Proof. The proof goes in the same way as Lemma 10. ∎

The right sides of the inequalities above have the following values for $k=3, 4, 5$ and 6. These values will be used later in the proof of the next theorem.

$$k=3: \tfrac{4}{5}(m-4)+\tfrac{5}{2}=\tfrac{4}{5}m-\tfrac{7}{10}$$
$$k=4: \tfrac{5}{6}(m-4)+\tfrac{13}{5}=\tfrac{5}{6}m-\tfrac{11}{15}$$
$$k=5: \tfrac{6}{7}(m-4)+\tfrac{8}{3}=\tfrac{6}{7}m-\tfrac{16}{21}$$
$$k=6: \tfrac{7}{8}(m-4)+\tfrac{19}{7}=\tfrac{7}{8}m-\tfrac{11}{14}$$

THEOREM 14. *For $k=1, 2, 3, 4, 5, 6$*

$$\omega^{Alg\ 6(FF)} \leqslant \left(2+\frac{1}{k+1}\right) \cdot \omega^*.$$

Proof. Here, Lemma 13 and Lemma 12 imply the following inequalities for $k=1, 2, 3, 4, 5, 6$:

$$k=1: \omega^* \geqslant \frac{a}{2} + \sum_{k_i=2} 1 + \sum_{k_i \geqslant 3} \left[\frac{2}{3}k_i - \frac{1}{3}\right]$$

$$k=2: \omega^* \geqslant \frac{a}{2} + \sum_{k_i=2} 1 + \sum_{k_i \geqslant 3} \left[\frac{3}{4}k_i - \frac{7}{12}\right]$$

$$k=3: \omega^* \geqslant \frac{a}{2} + \sum_{k_i=2} 1 + \sum_{k_i=3} \frac{7}{4} + \sum_{k_i \geqslant 4} \left[\frac{4}{5}k_i - \frac{7}{10}\right]$$

$$k=4: \omega^* \geqslant \frac{a}{2} + \sum_{k_i=2} 1 + \sum_{k_i=3} \frac{7}{4} + \sum_{k_i \geqslant 4} \left[\frac{5}{6}k_i - \frac{11}{15}\right]$$

$$k=5: \omega^* \geqslant \frac{a}{2} + \sum_{k_i=2} 1 + \sum_{k_i=3} \frac{7}{4} + \sum_{k_i \geqslant 4} \left[\frac{6}{7}k_i - \frac{16}{21}\right]$$

$$k=6: \omega^* \geqslant \frac{a}{2} + \sum_{k_i=2} 1 + \sum_{k_i=3} \frac{7}{4} + \sum_{k_i \geqslant 4} \left[\frac{7}{8}k_i - \frac{11}{14}\right]$$

Using the same method as in Theorem 6, we get as bound for $\omega^{\mathrm{Alg}\,5(FF)} = a + b + \sum_{k_i > 1} k_i$ the values $2\frac{1}{2}$, $2\frac{1}{3}$, $2\frac{1}{4}$, $2\frac{1}{5}$, $2\frac{1}{6}$, and $2\frac{1}{7}$ for $k = 1, 2, 3, 4, 5,$ and 6, respectively. ∎

The bound $2 + \frac{1}{7}$ can be achieved for Alg 6(FFD) and $k \geqslant 6$. This implies that the worst case ratio $\frac{15}{7}$ of Alg 6(FFD) cannot be improved using larger numbers $k \geqslant 7$. To prove this for $k = 6$ we construct the conflict graph $G^{(\ell)} = (J^{(\ell)}, E^{(\ell)})$ as follows. As job set $J^{(\ell)}$ we take the union of $\{a_{i,k} \mid 1 \leqslant i \leqslant 4, 1 \leqslant k \leqslant 4\ell\}$ and $\{b_k \mid 1 \leqslant k \leqslant 7\ell\}$ and as edge set $E^{(\ell)} = \{\{b_k, b_{k'}\} \mid 1 \leqslant k < k' \leqslant 7\ell\}$. The execution times of the jobs are

jobs	execution times
$a_{1,k}$	$\frac{3}{4} + k\varepsilon - \delta$
$a_{2,k}$ for odd $k \geqslant 3$	$\frac{1}{2} - 2^{k-1}\varepsilon - \delta$
$a_{2,k}$ for even k	$\frac{1}{2} - \delta$
$a_{2,k}$ for $k = 1$	$\frac{1}{2} - \delta$
$a_{3,k}$ for even k	$\frac{1}{4} + 2^{k/2}\varepsilon - \delta$
$a_{3,k}$ for odd $k \geqslant 3$	$\frac{1}{4} + [2^{k-1} + 2^{(k+1)/2}]\,\varepsilon - \delta$
$a_{3,k}$ for $k = 1$	$\frac{1}{4} + 2\varepsilon - \delta$
$a_{4,k}$	$\frac{1}{4} - (k-1)\varepsilon - \delta$
b_k	$\delta \ll \varepsilon$

An optimum coloring where each color set contains at most ℓ jobs $j \in J$ with execution times $e(j) \in (1/(\ell+1), 1]$ for $1 \leqslant \ell \leqslant 7$ is given by independent sets $U_k = \{a_{i,k} \mid 1 \leqslant i \leqslant 4\} \cup \{b_k\}$ for $1 \leqslant k \leqslant 4\ell$ and $U_k = \{b_k\}$ for $4\ell + 1 \leqslant k \leqslant 7\ell$. Notice that $G^{(\ell)}$ contains a clique of size 7ℓ. If we apply FFD on these color sets, we get $12\ell + 3\ell = 15\ell$ sets.

On the other hand, an optimum solution for the scheduling problem has $7\ell + 1$ sets:

job sets	indices	number of sets
$\{a_{1,k}, a_{4,k+1}, b_k\}$	$1 \leqslant k \leqslant 4\ell - 1$	$4\ell - 1$
$\{a_{3,2k-1}, a_{3,2k}, a_{2,2k+1}, b_{4\ell-1+k}\}$	$1 \leqslant k \leqslant 2\ell - 1$	$2\ell - 1$
$\{a_{2,4k-2}, a_{2,4k}, b_{6\ell-2+k}\}$	$1 \leqslant k \leqslant \ell$	ℓ
$\{a_{4,1}, a_{3,4\ell-1}, a_{3,4\ell}, b_{7\ell-1}\}$		1
$\{a_{1,4\ell}, b_{7\ell}\}$		1
$\{a_{2,1}\}$		1

In total, $\lim_{\ell \to \infty}(15\ell/(7\ell + 1)) = 2\frac{1}{7}$.

6. APPROXIMATION CLOSE 2

In this section, we propose an approximation algorithm to get the worst case ratio $2 + (1/(k+1))$ for any constants $k \in \mathbb{N}$. Using this method we obtain approximation algorithms with the worst case bound $2 + \varepsilon$ for cographs and partial K-trees.

ALGORITHM 7.

Step 1. Let $\bar{J}_0 = \{ j \in J \mid e(j) \in (\frac{3}{4}, 1]\}$, $\bar{J}_1 = \{ j \in J \mid e(j) \in (\frac{1}{2}, \frac{3}{4}]\}$ and $\bar{J}_\ell = \{ j \in J \mid e(j) \in (1/(\ell+1), 1/\ell]\}$ for each $2 \leqslant \ell \leqslant k$.

Step 2. Compute a minimum coloring of the confict graph G with independent sets $U_1, ..., U_m(G)$ where

$$\sum_{l=2}^{k} \frac{1}{\ell+1} |U_i \cap \bar{J}_\ell| + \frac{1}{2} |U_i \cap \bar{J}_1| + \frac{3}{4} |U_i \cap \bar{J}_0| < 1 \qquad (*)$$

for each $1 \leqslant i \leqslant m$.

Step 3. For each $1 \leqslant i \leqslant m$, place the jobs $j \in U_i$ with execution time $e(j) > 1/(k+1)$ into at most two sets A_i, B_i with execution times $e(A_i)$, $e(B_i) \leqslant 1$.

Step 4. For each $1 \leqslant i \leqslant m$, apply a bin-packing heuristic (*NF*, *FF*, *FFD*) to $U_i \backslash (A_i \cup B_i)$ where the sets A_i, B_i are placed before into the first and second bin.

LEMMA 15. *The jobs $j \in U_i$ with execution times $e(j) > 1/(k+1)$ can be placed into at most two bins.*

Proof. Let U be an independent set of jobs $j \in J$ with execution times $e(j) > 1/(k+1)$ where the inequality $(*)$ (see Step 2 of the algorithm) is satisfied.

We show that the sum $\sum_{j \in U} e(j) < \frac{3}{2}$. This inequality implies directly that we can separate the jobs into at most two sets A and B with execution times $e(A)$, $e(B) \leqslant 1$. Let $L = (j_1, ..., j_n)$ be a list of the jobs in U and let i be the smallest index such that $\sum_{\ell=1}^{i} e(j_\ell) \geqslant 1$. If $\sum_{\ell=1}^{i-1} e(j_\ell) \geqslant \frac{1}{2}$ then we set $A = \{ j_\ell \mid 1 \leqslant \ell \leqslant i-1\}$ and $B = U \backslash A$. Otherwise, if $\sum_{\ell=1}^{i-1} e(j_\ell) < \frac{1}{2}$ then $e(j_i) > \frac{1}{2}$ and we set $A = \{ j_i\}$ and $B = U \backslash A$.

Using the inequality $(*)$, the sum of the execution times can be bounded as follows:

$$\begin{aligned} \sum_{j \in U} e(j) &\leqslant \sum_{\ell=2}^{k} \frac{1}{\ell} |U \cap \bar{J}_\ell| + \frac{3}{4} |U \cap \bar{J}_1| + 1\, |U \cap \bar{J}_0| \\ &\leqslant \frac{3}{2} \left[\sum_{\ell=2}^{k} \frac{1}{\ell+1} |U \cap \bar{J}_\ell| + \frac{1}{2} |U \cap \bar{J}_1| + \frac{3}{4} |U \cap \bar{J}_0| \right] \\ &< \frac{3}{2} \cdot 1 = \frac{3}{2}. \quad \blacksquare \end{aligned}$$

LEMMA 16. *Let k be a positive integer and let L be a list of positive numbers $a_1, ..., a_n \leqslant 1$. Let $B_1, ..., B_m$ $(m \geqslant 3)$ be the set of non-empty bins generated by FF on L where the first two bins B_1 and B_2 are filled with all numbers $a_i > 1/(k+1)$. If b_i is the level of bin B_i, $1 \leqslant i \leqslant m$, then*

$$\sum_{i=1}^{m} b_i \geqslant \frac{k+1}{k+2} \cdot (m-3) + \frac{k}{k+1} + 1.$$

Proof. If $m = 3$, the last bin B_3 contains at least one number $y \leqslant 1/(k+1)$. Then, $b_1, b_2 > k/(k+1)$ and $b_1 + b_2 + b_3 > 1 + k/(k+1)$.

Now, let $m>3$. If all levels $b_3, ..., b_{m-1} \geqslant (k+1)/(k+2)$, then the inequality above is true. Therefore, we assume that there is an index j, $3 \leqslant j < m$ such that $b_j < (k+1)/(k+2)$. Choosing the smallest index j with this property, $b_{j'} \geqslant (k+1)/(k+2)$ for $k+1 \leqslant j' < j$ and $j < j' < m$. Since B_j contains at least $k+1$ numbers and $b_j < (k+1)/(k+2)$, there is one number $y \in B_j$ with $y < 1/(k+2)$. Therefore, we have $b_1, b_2 > (k+1)/(k+2)$ and

$$\begin{aligned}\sum_{i=1}^{m} b_i &> \frac{k+1}{k+2}(m-2)+1 \\ &\geqslant \frac{k+1}{k+2}(m-3)+\frac{k+1}{k+2}+1 \\ &\geqslant \frac{k+1}{k+2}(m-3)+\frac{k}{k+1}+1. \quad \blacksquare\end{aligned}$$

THEOREM 17. *For each integer $k \geqslant 1$:*

$$\omega^{Alg\ 7(FF)} \leqslant \left(2+\frac{1}{k+1}\right)\omega^*.$$

Proof. Again, we consider a partition into independent sets $U_1, ..., U_m$ where m is the minimum number of colors in a coloring such that each set U_i satisfies the inequality $(*)$. Lemma 15 implies that the jobs $j \in U_i$ with execution time $e(j) > 1/(k+1)$ can be placed into at most two sets. Therefore, we can apply Lemma 16 and get the following inequality for the optimum value ω^* of the scheduling problem:

$$\omega^* \geqslant \frac{a}{2} + \sum_{k_i=2} 1 + \sum_{k_i \geqslant 3} \left[\frac{k+1}{k+2}k_i - 3\frac{k+1}{k+2} + \frac{k}{k+1} + 1\right].$$

This inequality can be transformed into

$$\begin{aligned}\sum_{k_i \geqslant 3} k_i \leqslant & \frac{k+2}{k+1}\omega^* - \frac{k+2}{2(k+1)}a - \sum_{k_i=2}\frac{k+2}{k+1} \\ &+ \sum_{k_i>2}\left[3 - \frac{k(k+2)}{(k+1)^2} - \frac{k+2}{k+1}\right].\end{aligned}$$

The algorithm Alg 7(FF) generates at most

$$\begin{aligned}\omega^{\text{Alg } 7(FF)} &= a + b + \sum_{k_i=2} 2 + \sum_{k_i>1} k_i \\ &\leqslant \frac{k+2}{k+1}\omega^* + a\left[1 - \frac{k+2}{2(k+1)}\right] + b + \sum_{k_i=2}\left[2 - \frac{k+2}{k+1}\right] \\ &\quad + \sum_{k_i>2}\left[3 - \frac{2k^2+5k+2}{k^2+2k+1}\right].\end{aligned}$$

Since

$$1-\frac{k+2}{2(k+1)}<1, \qquad 2-\frac{k+2}{k+1}<1,$$

and

$$3-\frac{2k^2+5k+2}{k^2+2k+1}<1 \qquad \text{for each} \quad k\geqslant 1,$$

we obtain

$$\begin{aligned}\omega^{\text{Alg }7(FF)} &\leqslant \frac{k+2}{k+1}\,\omega^*+m\\ &\leqslant \frac{k+2}{k+1}\,\omega^*+\omega^*\\ &=\left(2+\frac{1}{k+1}\right)\omega^*. \quad \blacksquare\end{aligned}$$

COROLLARY 18. *The algorithm Alg 7(FF) runs in polynomial time and has the absolute worst case ratio $2+\varepsilon$ on the following graph classes: cographs and partial K-trees for constant K.*

7. CONCLUSION

In this paper, we studied the problem of scheduling a set J of jobs with execution times $e(j)\in[0,1]$ and a conflict graph $G=(J,E)$. We were searching for a partitioning of the job set J into independent sets $U_1, ..., U_m$ such that $\sum_{j\in U_i} e(j)\leqslant 1$ for each $1\leqslant i\leqslant m$ and m should be minimal. Set U_i is executed on one machine.

Investigating different methods for approximative solutions to this problem we were able to show the results summarized in Table 1. There, ω^H is the number of machines required when the algorithm H is used, and ω^* is the minimal number of machines needed.

TABLE I

Summary of the approximation methods

Algorithm H	ω^H
Direct bin-packing A_1	$\omega^{A_1}=\mathcal{O}(\lvert J\rvert)\cdot\omega^*$
Coloring A_2	$\omega^{A_2}\leqslant 2.7\cdot\omega^*$
Precoloring A_3	$\omega^{A_3}\leqslant 2.5\cdot\omega^*$
General Coloring A_4	$\omega^{A_4}\leqslant(2+\frac{1}{3})\cdot\omega^*$
General Coloring A_5	$\omega^{A_5}\leqslant(2+\frac{1}{5})\cdot\omega^*$
General Coloring A_6	$\omega^{A_6}\leqslant(2+\frac{1}{7})\cdot\omega^*$
2-Approximation A_7	$\omega^{A_7}\leqslant(2+\varepsilon)\cdot\omega^*$

The following problems are interesting for further research:

(1) Complexity of $1 - PrExt$ for other graph classes (e.g., chordal graphs).

(2) Approximation algorithms for special graph classes.

(3) Other separation methods to distribute the jobs with long execution times.

(4) ε approximability and/or MAX-SNP hardness of the problem for different graph classes.

Received July 26, 1996; final manuscript received August 7, 1996

REFERENCES

1. Baker, B. S., and Coffman, E. G. (1996), Mutual exclusion scheduling, *Theor. Comput. Sci.* **162**, 225–243.
2. Biró, M., Hujter, M., and Tuza, Z. (1991), Precoloring extension, I: Interval graphs, *Discrete Math.* **100**, 267–279.
3. Bodlaender, H. L., Jansen, K., and Woeginger, G. J. (1992), Scheduling with incompatible jobs, *in* "Workshop Graph Theoretical Concepts in Computer Science," LNCS, Vol. 657, pp. 37–49, Springer-Verlag, Berlin/New York.
4. Bodlaender, H. L., and Jansen, K. (1993), On the complexity of scheduling incompatible jobs with unit-times, *in* "Mathematical Foundations of Computer Science," LNCS, Vol. 711, pp. 291–300, Springer-Verlag, Berlin/New York.
5. Brandstädt, A. (1991), "Special Graph Classes—A Survey," Report SM-DU-199, Universität-Gesamthochschule Duisburg, 1991.
6. Coffman, E. G. Jr., Garey, M. R., and Johnson, D. S. (1984), Approximation algorithms for bin-packing—An updated survey, *in* "Algorithmic Design for Computer System Design" (G. Ausiello, M. Lucertini, and P. Serafini, Eds.), pp. 49–106, Springer-Verlag, Berlin/New York.
7. Garey, M. R., Graham, R. L., Johnson, D. S., and Yao A.C.-C. (1976), Resource constrained scheduling as generalized bin-packing, *J. Combin. Theory Ser. A* **21**, 257–298.
8. Garey, M. R., and Johnson, D. S. (1979), "Computers and Intractability: A Guide to the Theory of NP-Completeness," Freeman, San Francisco, 1979.
9. Golumbic, M. C. (1980), "Algorithmic Graph Theory and Perfect Graphs," Academic Press, London.
10. Hujter, M., and Tuza, Z. (1993), Precoloring extension, II: Graph classes related to bipartite graphs, *Acta Math. Univ. Comenianae* **61**, 1–12.
11. Hujter, M., and Tuza, Z. (1996), Precoloring extension, III: Classes of perfect graphs, *Combin. Probab. Comput.* **5**, 35–56.
12. Irani, S., and Leung, V. (1996), Scheduling with conflicts, and applications to traffic signal control, *in* "Symposium on Discrete Algorithms," pp. 85–94.
13. Jansen, K., and Scheffler, P. (1992), Generalized coloring for tree-like graphs, *in* "Workshop Graph Theoretical Concepts in Computer Science," LNCS, Vol. 657, pp. 50–59, Springer-Verlag, Berlin/New York.
14. Johnson, D. S., Demers, A., Ullman, J. D., Garey, M. R., and Graham, R. L. (1974), Worst-case performance bounds for simple one-dimensional packing algorithms, *SIAM J. Comput.* **3**, 256–278.
15. Kratochvil, J. (1993), Precoloring extension with fixed color bound, *Acta Math. Univ. Comenianae* **62**, 139–153.

16. Lund, C., and Yannakakis, M. (1993), On the hardness of approximating minimization problems, *in* "25th Symposium on the Theory of Computing, 1993," pp. 286–293.

17. Minyi, Y. (1991), A simple proof of the inequality $FFD(L) \leqslant 11/9\ OPT(L) + 1$, $\forall L$ for the *FFD* bin-packing algorithm, *Acta Math. Appl. Sinica* **7**, 321–331.

18. Simchi-Levi, D. (1994), New worst-case results for the bin-packing problem, *Naval Res. Logistics*, **41**, 579–585.